

CMPE 540 Assignment #3

“A GGP Client”

Problem Statement

General Game Playing (GGP) Systems are systems that accept a formal description of a game and play the game effectively without human intervention. They are characterized by their use of AI techniques such as knowledge representation, reasoning, learning, and rational behavior. Unlike specialized game playing systems (such as Deep Blue), they do not rely exclusively on algorithms designed in advance for specific games like chess.

The underlying technology can also be used in a variety of other application areas, such as enterprise management, electronic commerce, security, and military operations. (See <http://games.stanford.edu/overview.html>)

For this project you can work in groups of two. You are to design a GGP client that is able to communicate with Stanford's game server (<http://games.stanford.edu:4000/>) and play multiplayer games effectively. You can find the necessary information in <http://games.stanford.edu/> or in this GGP course website: http://www.inf.tu-dresden.de/?node_id=1378&ln=en . Make sure that you go through lecture notes 1, 2 and 7 on the latter website. Also the official description of the system and its language GPL can be found in http://games.stanford.edu/gdl_spec.pdf .

The game server will send the description of the game to the clients in the beginning of each game. Along with this description, the amount of time you will have initially and in between the turns will be sent as STARTCLOCK and PLAYCLOCK variables. Your player should “think” before the game starts to “understand” the game and design any heuristics if possible. STARTCLOCK is the amount of time that can be used for this purpose.

After you choose and send a legal action, the game server will send you the move back, which you should compare with your intended move. A mismatch occurs if the action could not be successfully sent, or if you didn't provide a legal action. Your program must be able to choose a legal action, and update the game states accordingly. You have PLAYCLOCK amount of time to think between the turns.

You can use as a basis the LISP player given in Stanford's website, or the Java program given in Dresden's website. Any other client you find on the internet, you should first check with the assistant (keskinc@cmpe.boun.edu.tr or sys64738@gmail.com).

For multiplayer games you should consider your opponents possible moves while thinking, and update the game state accordingly when your opponent makes a move.

What to do

You should design and develop a GGP client that is able to communicate with Stanford's game server and plays multiplayer games effectively. In order to make debugging and testing possible, your client should print out each move it played, each move sent back by the game server, any pattern you have found, any heuristic your player could come up with, etc. on the screen, and to a text file. Remember that reports are very important parts of your projects.

Bonus: Find a way of making your client "remember" the games it played, and learn a playing method. You can model your opponent and try to guess its upcoming movements while thinking. You can use any game theory aspect for this purpose. There will be an additional bonus if you use neural networks to model games or players.

What to Submit

- a) A hardcopy report that explains all the algorithms you have chosen to implement the client specifically the ones that search the game state space. You should also present test results from several multiplayer games in Stanford's game manager with relevant discussion.
- b) Attached to the report a CD containing
 - a. your source codes,
 - b. sample game transcripts

Deadline

January 20, 2007 Saturday 17:00